CSN08x14

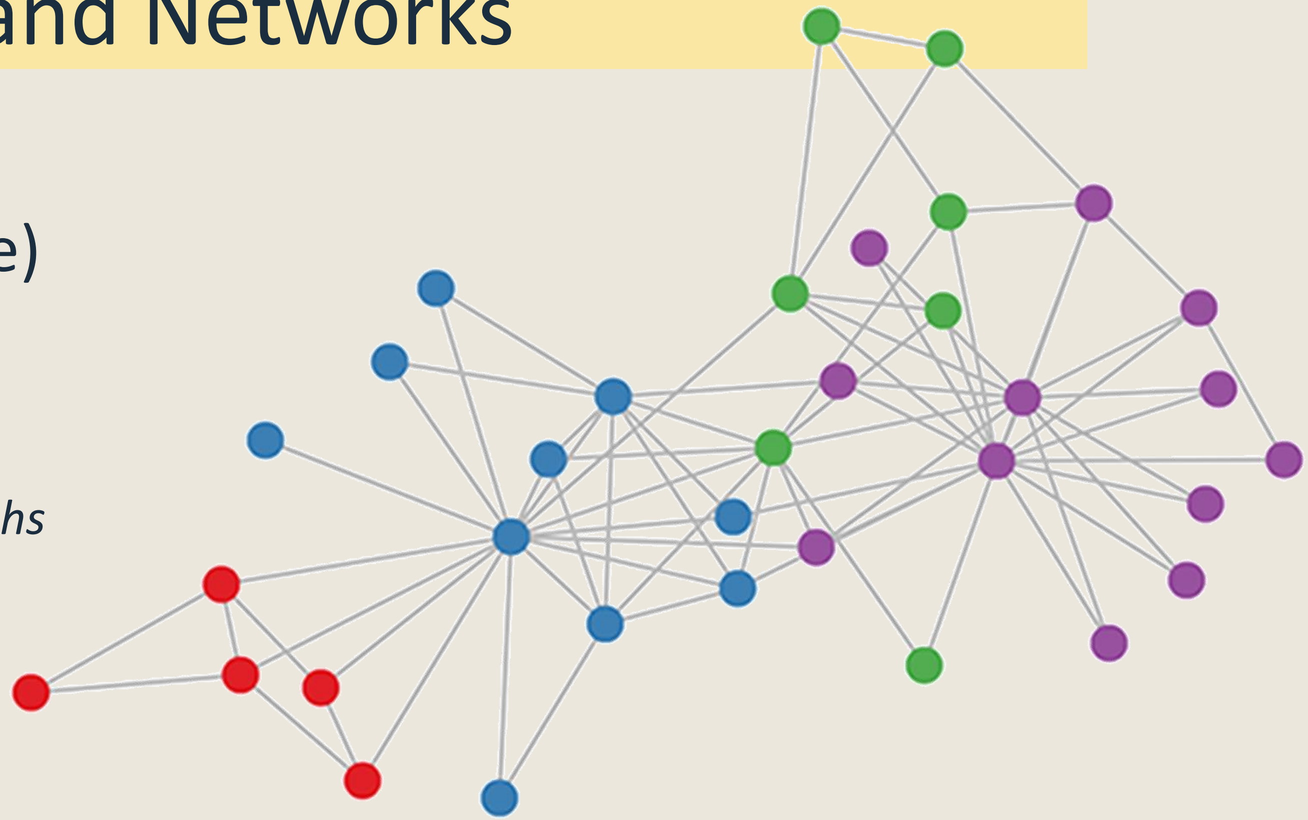
Scripting for Cybersecurity and Networks

Lecture 10:

Graphs and Networks

In this lecture: Graphs (Graph Theory) and Networks

- The origin of graph theory
- Graphs (in the graph theory sense)
  - *Nodes and edges*
  - *Undirected and directed graphs*
  - *Complete, connected, unconnected graphs*
- Eulerian and Hamiltonian paths
- Applications
- The small world effect and small world networks
  - *Diameter and radius*
  - *Clustering coefficient*
  - *Examples: Scale-free and Watts & Strogatz networks*



Go to www.menti.com

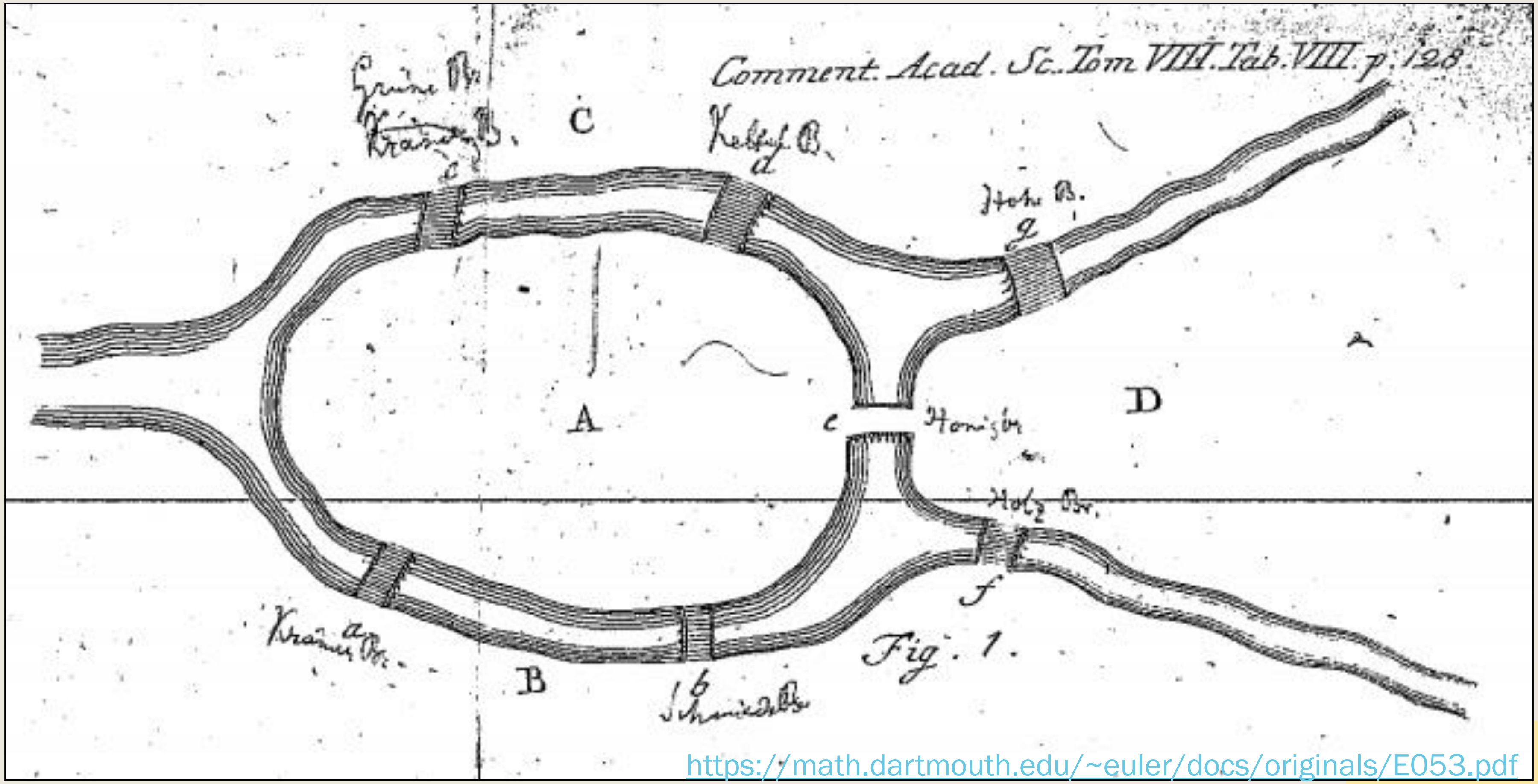
Code 34 89 26

We will continue to discuss the coursework at the end of this lecture (separate slides)

Origins of graph theory
Applications in Cybersecurity and Computing

The 7 bridges of Königsberg

- Sunday walk crossing each bridge exactly once?
- Try it! http://gwydir.demon.co.uk/jo/games/puzzles/bridge.htm.



https://math.dartmouth.edu/~euler/docs/originals/E053.pdf

Topology - Graph theory

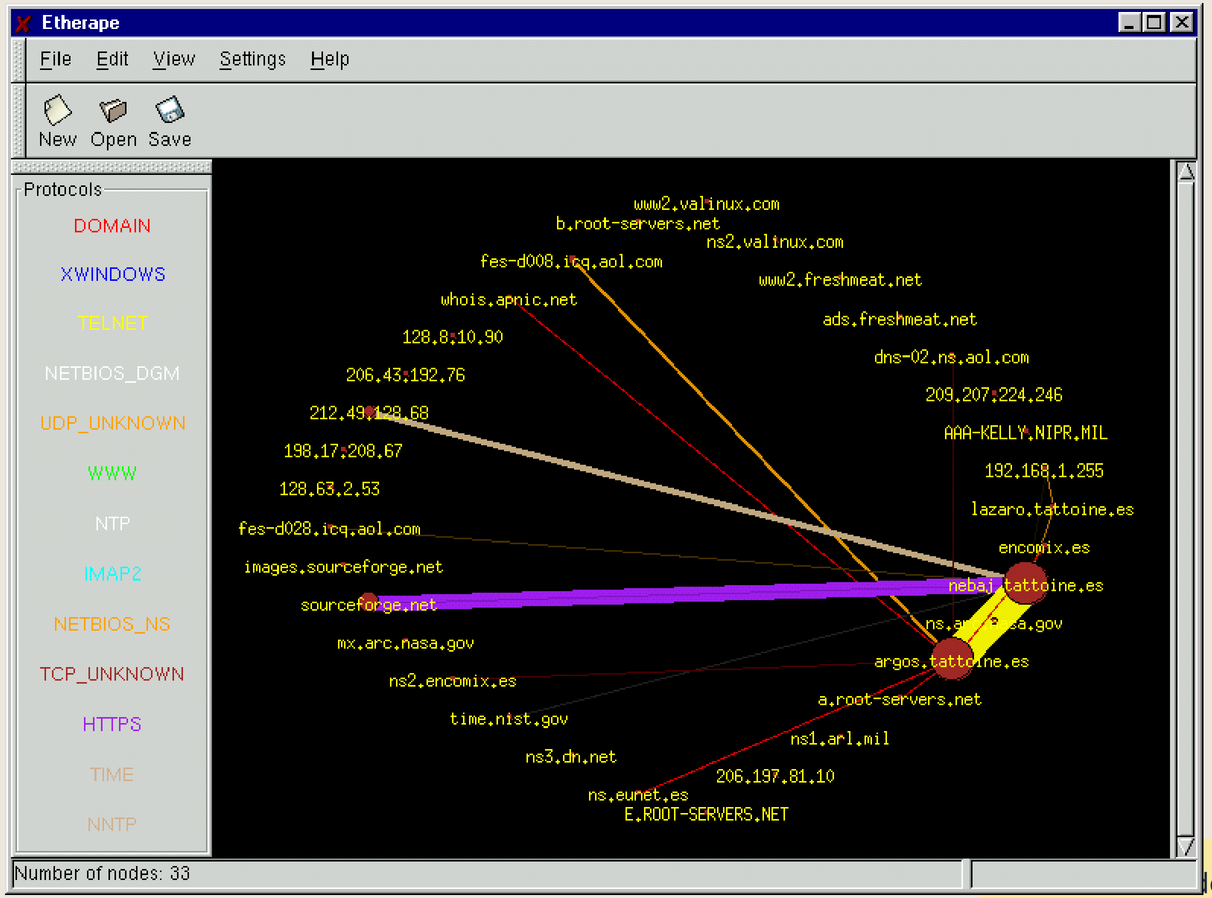
- Euler proved in 1735 that there is no solution to the 7 bridges of Königsberg problem (http://en.wikipedia.org/wiki/Seven_Bridges_of_K%C3%B6nigsberg; then in East Prussia, now Kaliningrad in Russia)

- The beginning of graph theory or topology

Application: Cybersecurity

- Many computer security tools use graphs for visualisation
- Example: EtherApe network traffic monitoring tool (use with pcap) (http://etherape.sourceforge.net/)

- size of nodes and thickness of edges are proportional to traffic volume.
- edge colour denotes the prevalent protocol of the associated traffic.



Application: Chat-log analysis

- Anwar & Abulaish (2014)
- method to identify digital evidence from chat log data
- 3 scenarios for user-group identification

Tarique Anwar, Muhammad Abulaish

A social graph based text mining framework for chat log investigation

Digital Investigation,
Volume 11, Issue 4, 2014, 349–362

http://dx.doi.org/10.1016/j.diin.2014.10.001

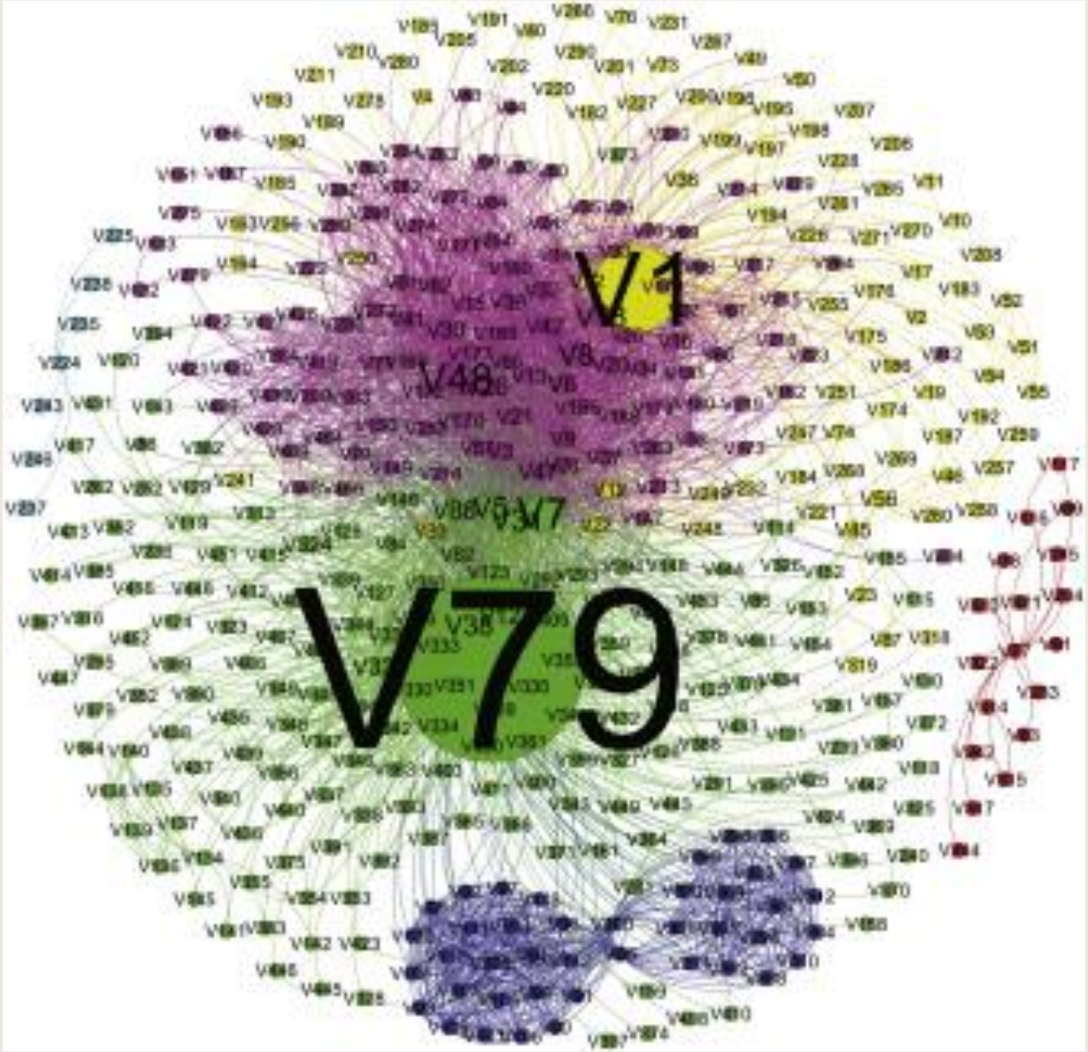


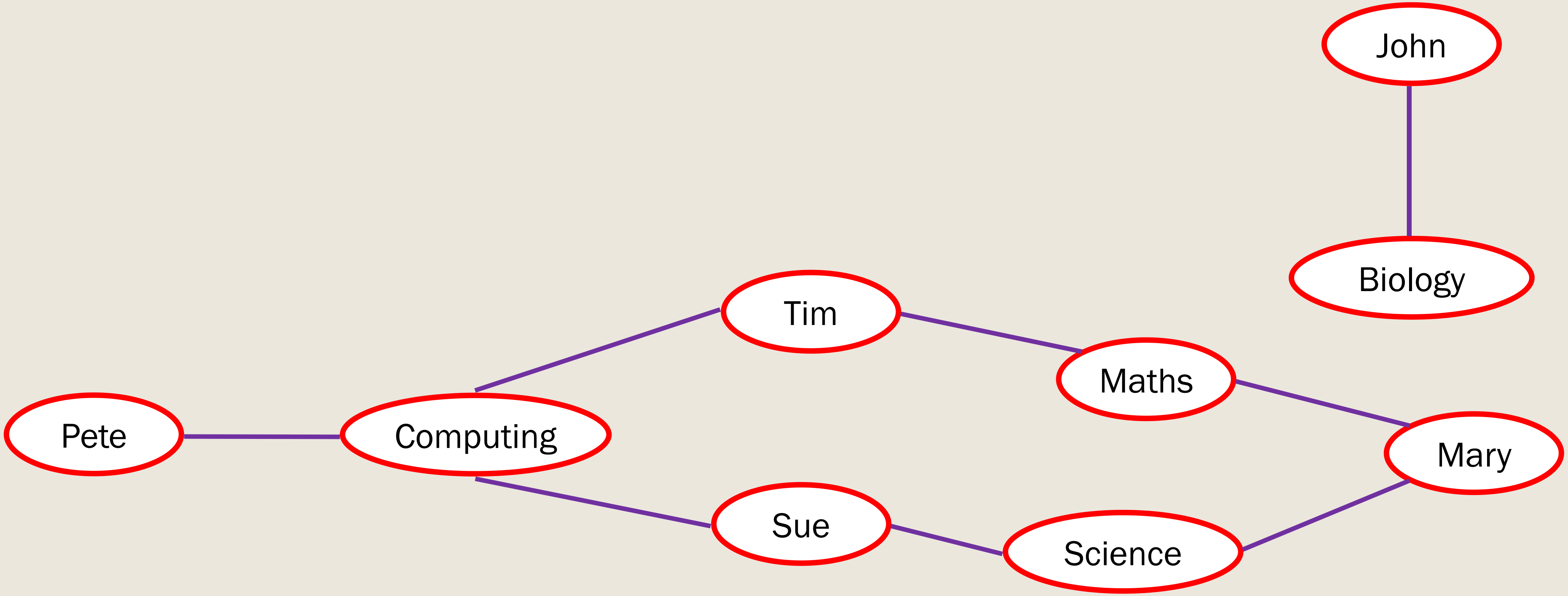
Fig. 5. A snapshot of the generated social graph.

Other graph applications in computing

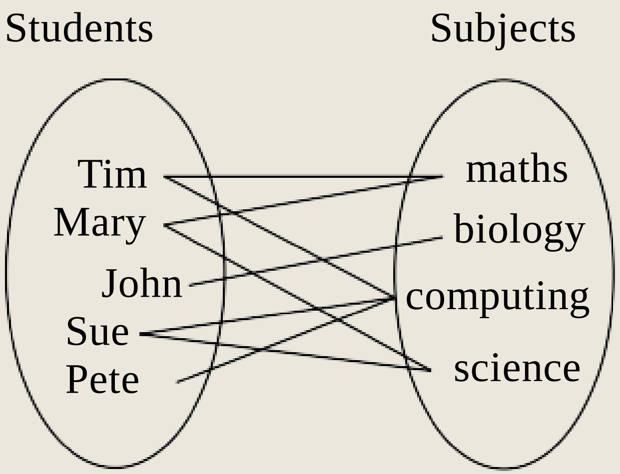
- Links between webpages
- Web site paths traversed by users
- Sitemaps
- Flow charts, UML diagrams
- Database schemata, ER diagrams
- Class hierarchies
- XML tree structures and DTDs
- Dependency diagrams / call graphs in programming
- Traveling salesman problem

Graph types & properties

A graph consists of nodes (vertices) and edges

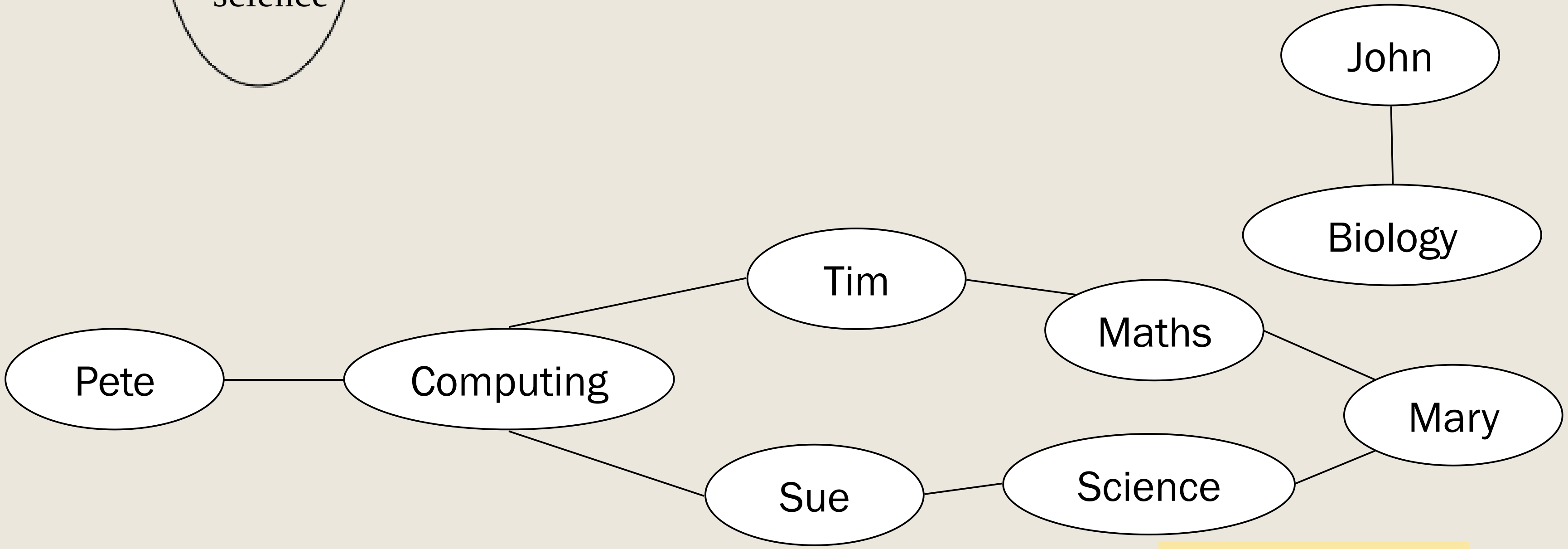


Graph to represent a binary relation



[(Tim, Computing), (Tim, Maths), (Mary, Maths), (Mary, Science), (John, Biology), (Sue, Computing), (Sue, Science), (Pete, Computing)]

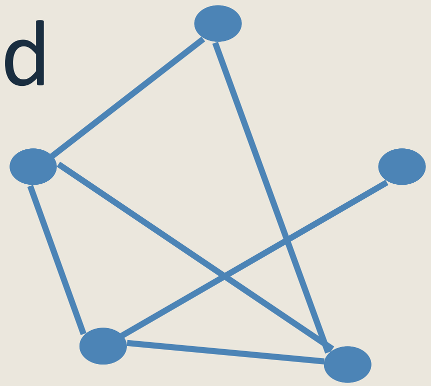
This is how we define a graph for using with NetworkX in Python



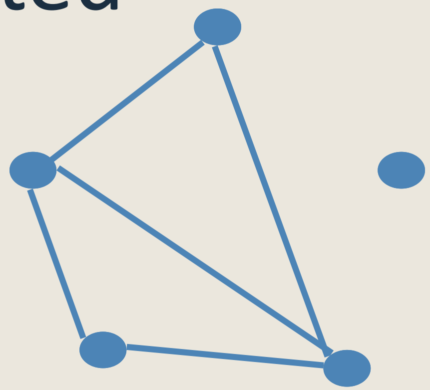
Connected, disconnected, complete graphs

A graph can be …

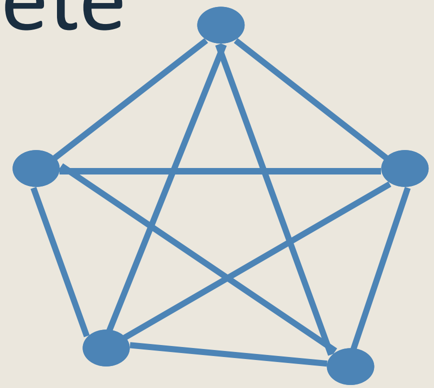
connected



disconnected



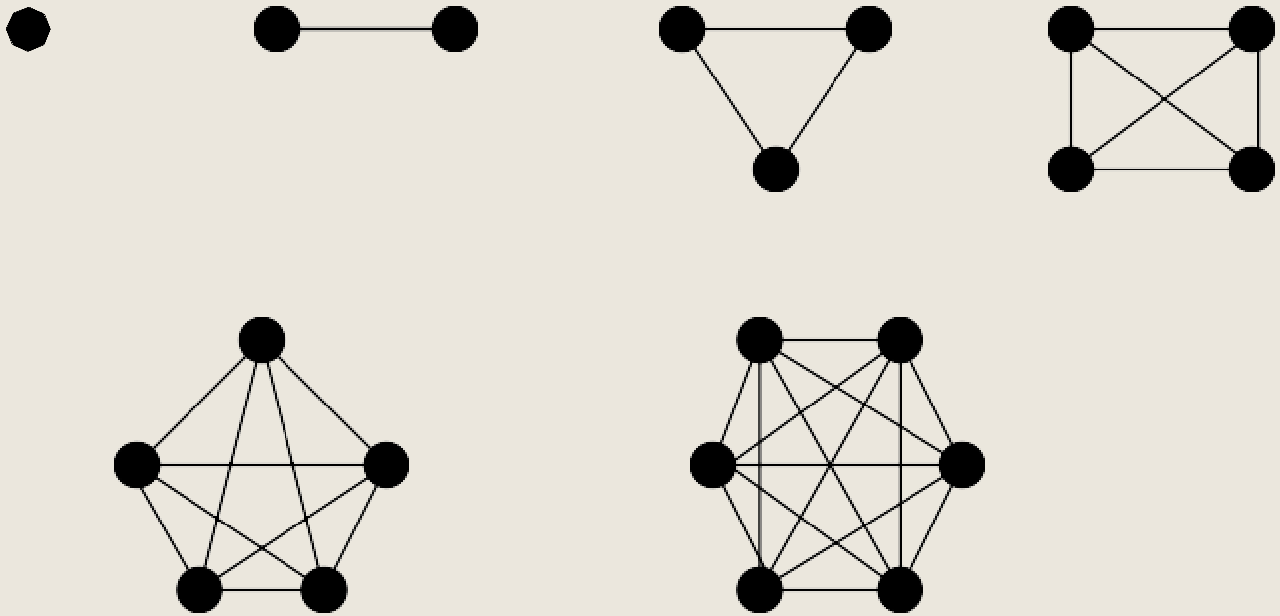
complete



Picture of a Null Graph

Complete graphs

- In a complete graph, every node is connected directly to every other node by an edge
- Complete graphs for n ≤ 6



Draw the complete graph with 7 nodes.

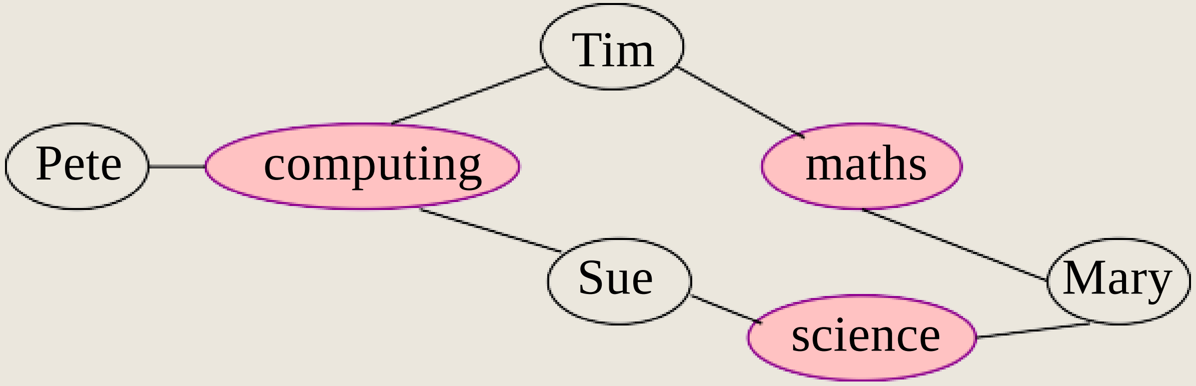
In a complete graph with n nodes…

…How many edges does every node have?
…How many edges does the whole graph have?

Bipartite / tripartite graphs

- A **bipartite** graph has two sets of nodes. A tripartite graph has 3.
- Edges are from one set of nodes to the other(s).
- There are no edges linking nodes within the same set

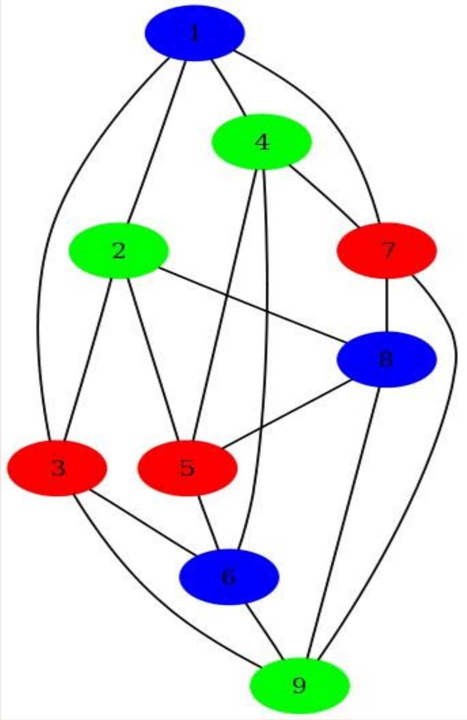
"Studies" graph with students and subjects



3x3 Sudoku

1	2	3
2	3	1
3	1	2

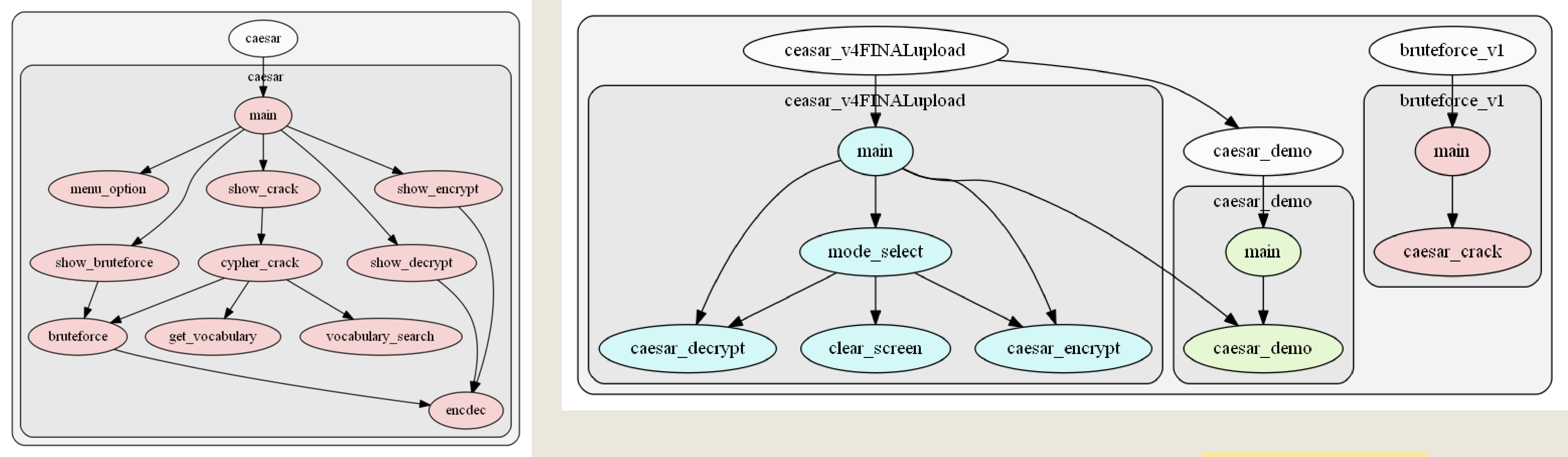
1	2	3
4	5	6
7	8	9



Q: Are there any complete graphs which are bipartite?

Directed graphs (digraphs)

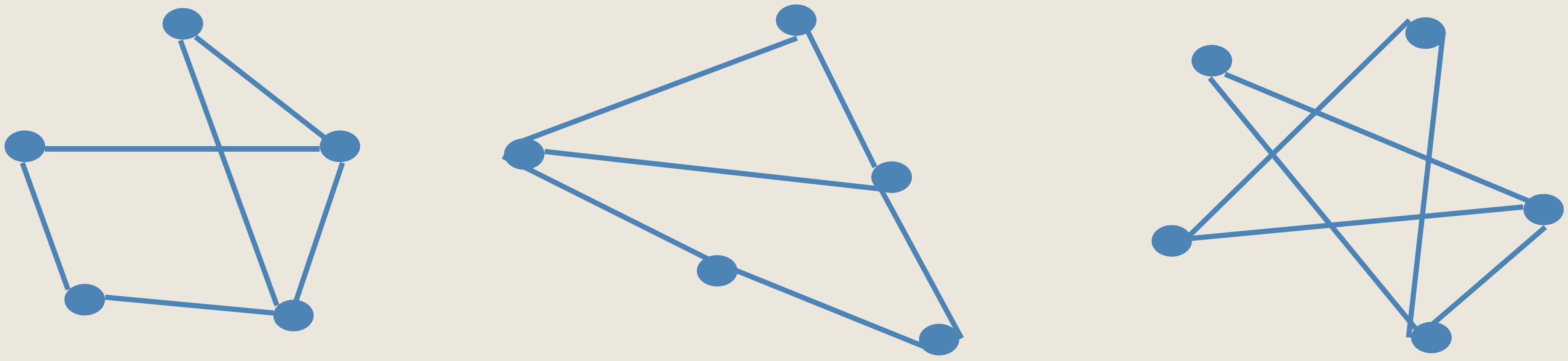
- Edges have a direction
- E.g. Functional dependency in code (also called call graphs or dependency diagrams)



Two dependency diagrams for a Caesar Cipher implementation in Python

Isomorphic graphs

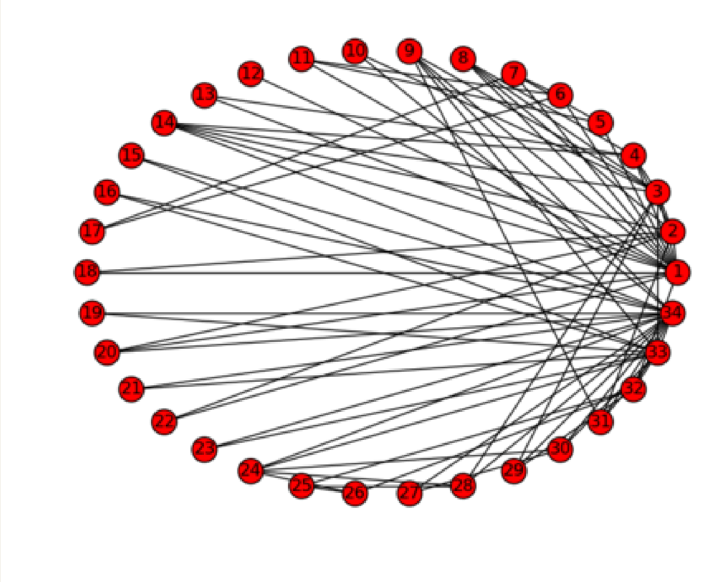
- Two graphs are isomorphic if they are "effectively the same"
  - *Same number of components (vertices and edges)*
  - *Same edge connectivity*
- Really just different representations/visualisations of the same graph
  - *One may be a much better visualisation than another*

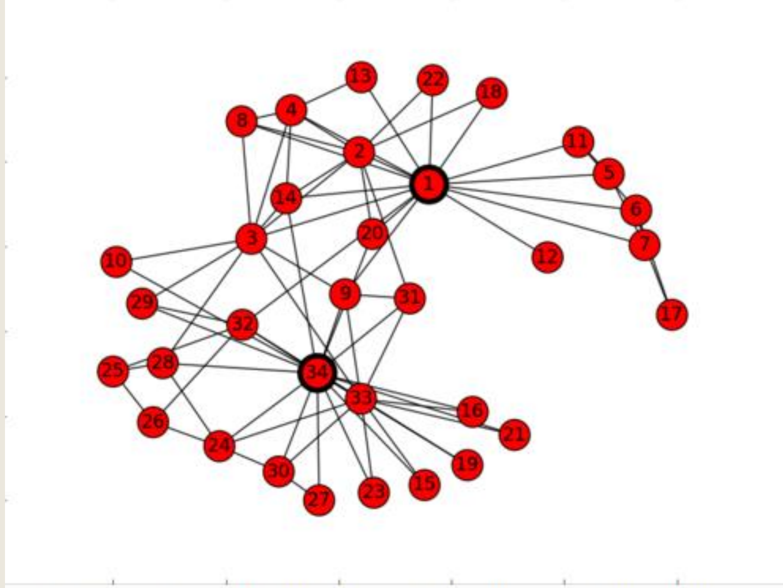


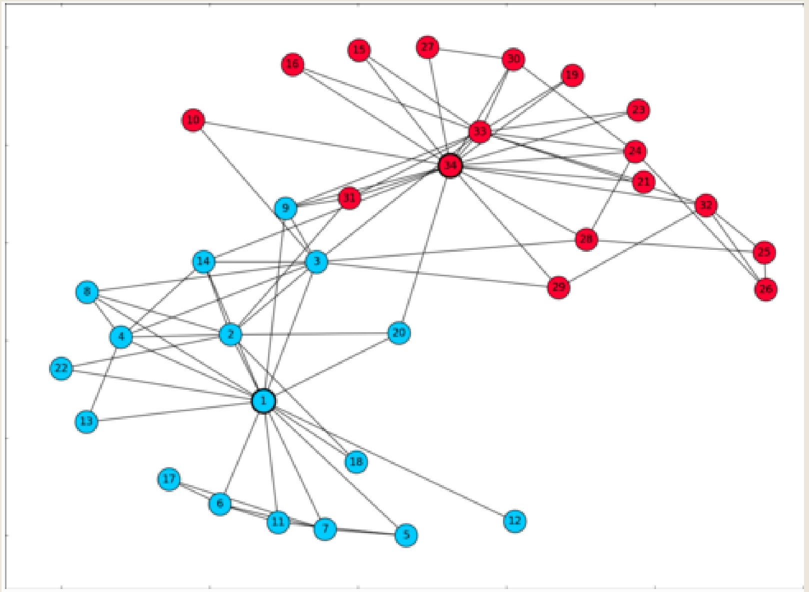
Three isomorphic graphs

Visualising graphs

- Different standard layouts – many algorithms
- Chosen layout can be very important


Zachary's karate club network: View 1 (Circular layout)


Zachary's karate club network: View 2 (Fruchterman-Reingold layout)


Zachary's karate club network: The 2 factions

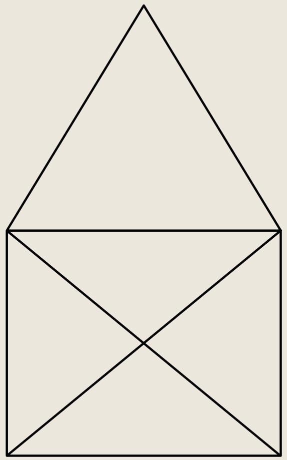
Graphs from http://www-rohan.sdsu.edu/~gawron/python_for_ss/course_core/book_draft/Social_Networks/Social_Networks.html.

Graphs tools in Python

- Several python packages
  - *NetworkX (https://networkx.github.io/)*
  - *igraph (http://igraph.org/python/)*
  - *Use with a suitable python drawing package, eg matplotlib*

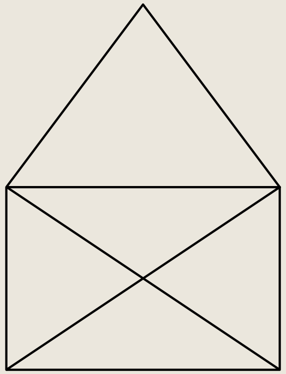
Can you draw this figure ...

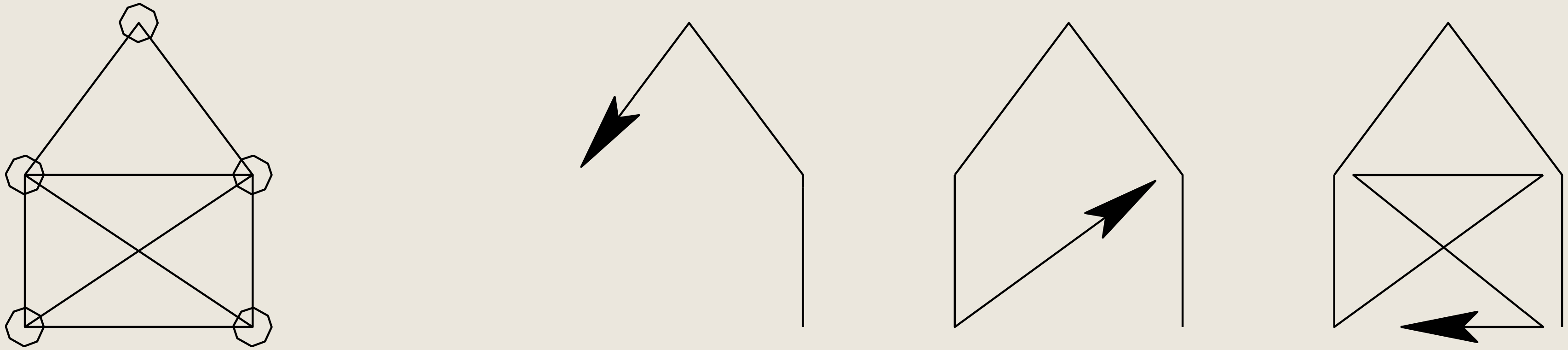
...in one go, without lifting your pen?



Can you draw this figure ...

...in one go, without lifting your pen?



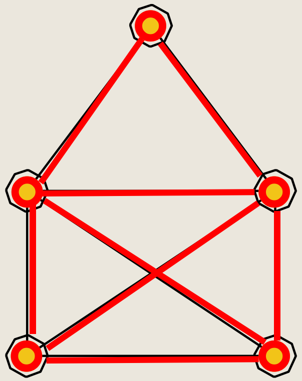


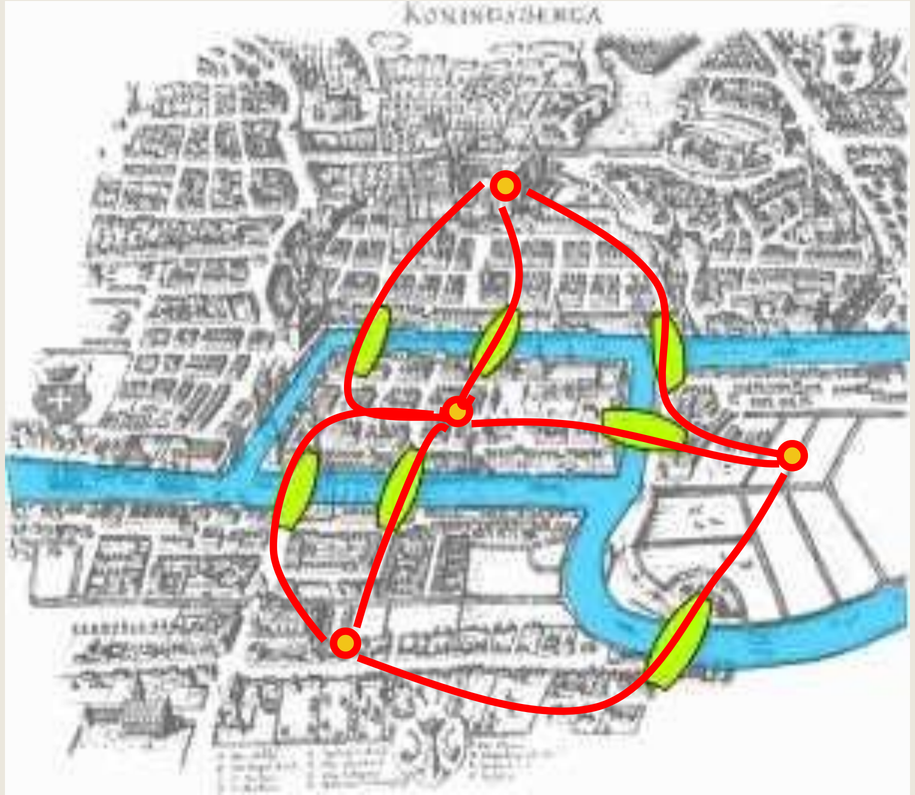
Eulerian path

- A path through a graph where each edge is visited exactly once.
- Eulerian circuit or cycle – an Eulerian path which starts and ends on the same node
- Euler asserted that a connected graph has an Eulerian path if and only if it has either no or two nodes with an odd number of edges.
  - *To have an Eulerian circuit, all nodes must have an even number of edges.*

Eulerian path?

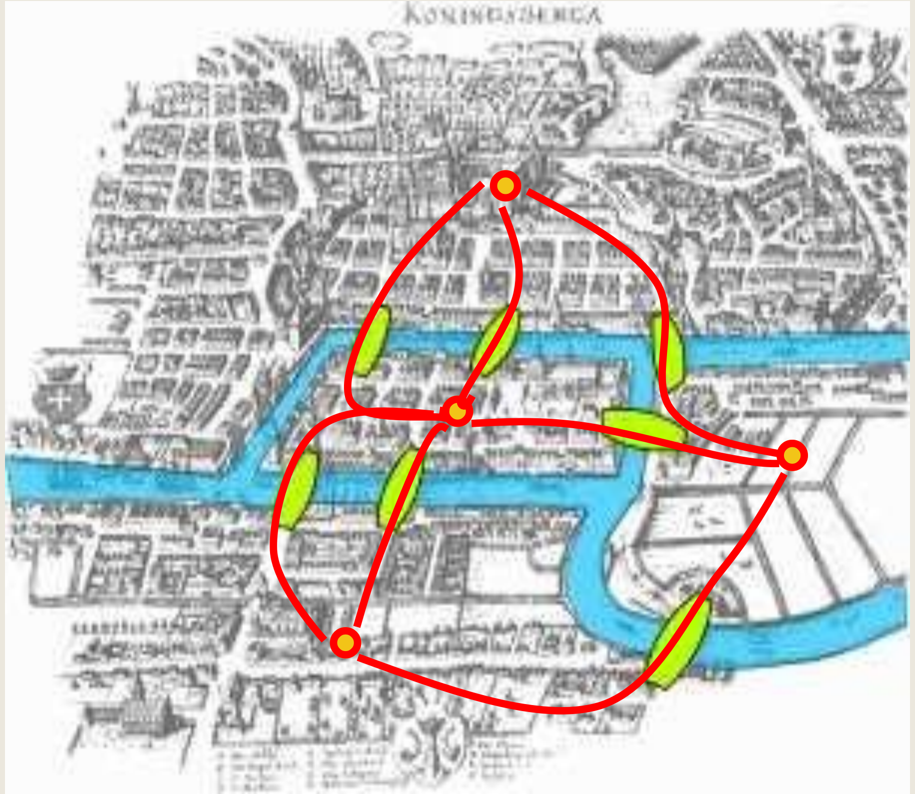
- How many nodes of odd degree does each of these graphs have?





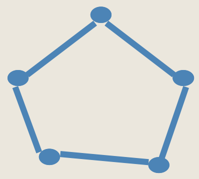
The seven bridges

- Because this graph has four edges of odd degree, it does not have an Eulerian path or circuit.
- This is how Euler proved that a Sunday walk crossing each bridge exactly once is impossible

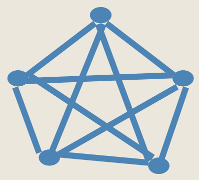


Hamiltonian path

- A path through a graph where each node is visited exactly once.
  - *E.g. cities in travelling salesman problem*
- Hamiltonian cycle: each node visited exactly once, finish in start node

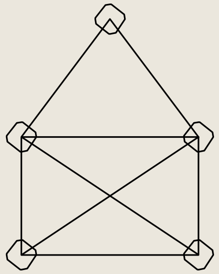


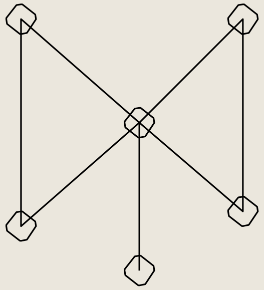
- All cycle graphs have Hamiltonian cycles
- All complete graphs with more than two nodes have Hamiltonian cycles



Hamiltonian path

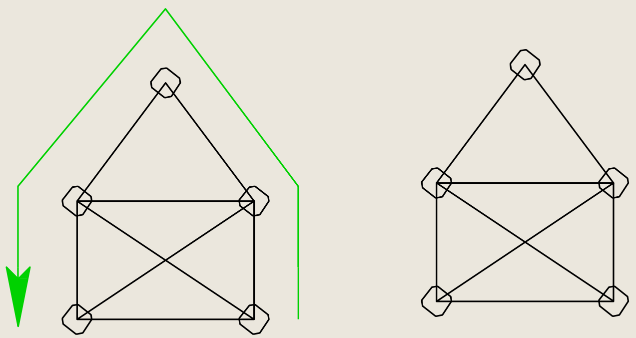
- Which of these graphs has a Hamiltonian path?



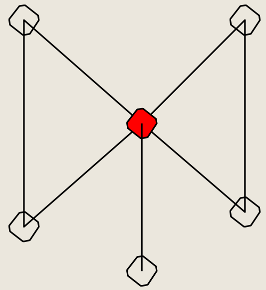


Hamiltonian path

- Which of these graphs has a Hamiltonian path?



(this one
– it has a Hamiltonian cycle)



(this one does not -
Impossible to visit every node without
passing through the red one more than once)

Application: Travelling salesman problem

- What is the shortest path for a salesman to visit a given set of cities?
- A classic problem of optimisation and planning
- A graph where the edges are labelled with the distances between the cities (weights)
- Among all the Hamiltonian paths, find the one which minimises distances.

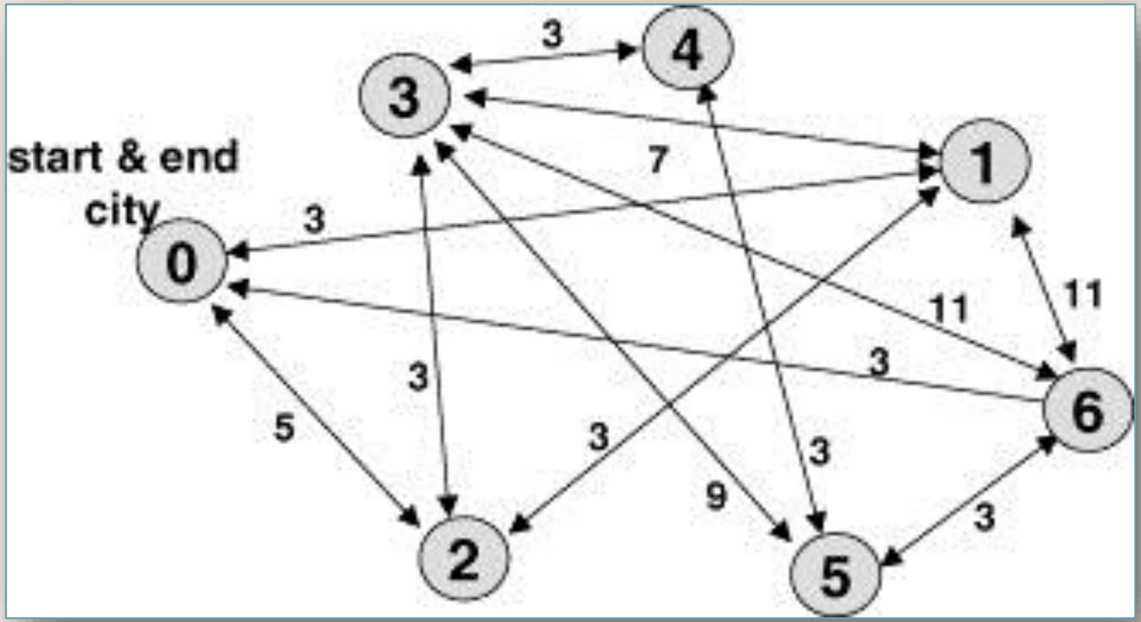
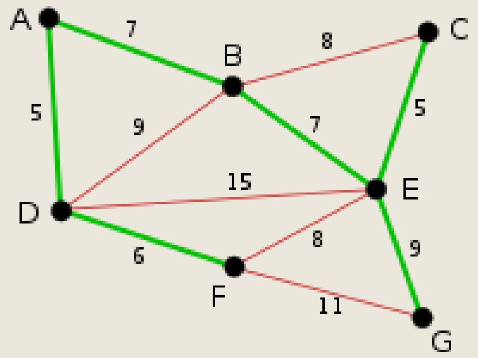


Image credit https://ars.els-cdn.com/content/image/1-s2.0-S0303264704001157-gr1.jpg

Minimum Spanning Tree MST

- Given a connected graph find the smallest subgraph that connects all the nodes.
- The subgraph will be a tree - it cannot have a loop
- The MST is a core – a minimal network required to keep all nodes connected.
- E.g.
  - *Minimum cables required in a (wired) computer network*
  - *Minimum of Roads/railway lines required to keep all nodes (towns, factories…) accessible*
- Kruskal's algorithm to find MST



Minimum spanning tree (green) from Wikipedia

Kruskal’s Algorithm for finding MST

- Sort the edges so the minimum weighted edge comes first
- For each edge
  - *If both end nodes are already in the MST then skip it*
  - *Otherwise add the current edge to the MST*
- Stop processing edges as soon as all the nodes are included

Scripting for Cybersec & Networks

kruskal1.py - F:\Dropbox\CSN08115_MADC\Petra_notes\kruskal1.py (3.6.1)

File Edit Format Run Options Window Help

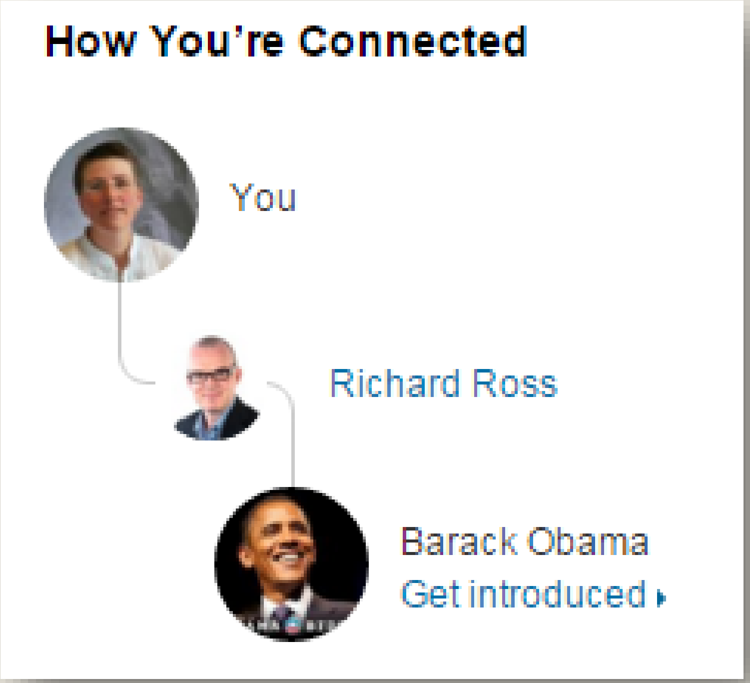
```python
# Kruskal's algorithm for finding Minimum Spanning Tree
# for Graphs lecture
N = 7 # number of nodes
# E contains edges in the format [node, node, weight]
E = [[1, 2, 7], [1, 4, 5], [2, 3, 8], [2, 4, 9],
     [2, 5, 7], [3, 5, 5], [4, 5, 15], [4, 6, 6],
     [5, 6, 8], [5, 7, 9], [6, 6, 11] ]

# Sort the edges, shortest first
E.sort(key=lambda a: a[2])
# Set of nodes that are in the mst
included = set() # ensures nodes can only be added once
# List of edges in MST
mst = []
while len(included)<N:
  h,E = E[0],E[1:] # h is the first edge
  print(f'checking edge {h}, nodes alredy added: {included}')
  print(f'edges remaining {E}')
  # Ignore edges where both nodes are included
  if h[0] in included and h[1] in included:
    pass # Do nothing - this is already there
  else:
    # Add both vertices to set of nodes
    included.add(h[0])
    included.add(h[1])
    mst.append(h) # add edge to mst
print (f'MST is {mst}')
```

Networks (Network Theory)

Computer networks are only one example of many different types of networks…

6 degrees of separation

- Do you know the US president?
- Do you know someone who knows the US president?
- Do you know someone who knows someone who knows the US president?
- ...
- The claim: everybody is connected to everybody else by at most 6 degrees of separation
  ⇒ It's a small world.



- LinkedIn shows 1st (direct), 2nd, 3rd degree connections (though no longer this particular view)
- …according to LinkedIn, I'm only 2 steps away from Barack Obama…
- …but at least 4 steps away from Hilary Clinton…

6 degrees of separation

- Idea from 1950's
- the World is becoming increasingly interconnected
- Is there any truth in the "6 degrees" idea?
  - *Stanley Milgram's Small-World Experiment*
  - *The Oracle of Bacon (the six degrees of Kevin Bacon)*
  - *The Erdös number*

Stanley Milgram's Small-World Experiment (1967)

- Sending packages to a stockbroker in Boston by sending them to random people in Nebraska and asking them to forward to someone who might know the stockbroker.
- Average path length around 5.5-6
- But several critiques
- https://en.wikipedia.org/wiki/Small-world_experiment.

The six degrees of Kevin Bacon

The Kevin Bacon game:

- start with any actor or actress who has been in a movie and connect them to Kevin Bacon in the smallest number of links possible (using the internet movie database).
- Two people are linked if they've been in a movie together.
- Bacon numbers higher than 4 are very rare
- Play the game at http://oracleofbacon.org/.

Erdös number

- Mathematician Paul Erdös published more papers than any other mathematician in history and had hundreds of co-authors.
- The Erdös number describes the "collaborative distance" between him and another person, as measured by authorship of mathematical papers.
- Co-authors have Erdös number 1.
- Co-authors of co-authors have Erdös number 2.
- 90 percent of the world's active mathematicians have an Erdös number smaller than 8.
- It has been said: Someone who does not have an Erdös number is not a mathematician.

http://wwwp.oakland.edu/enp/, https://en.wikipedia.org/wiki/Erd%C5%91s_number.

Small-world effect

- Any node can be reached from any other node by a short path.
- i.e. small graph diameter
- Examples:
  - *Social networks*
  - *Internet*
  - *Electric power grids*

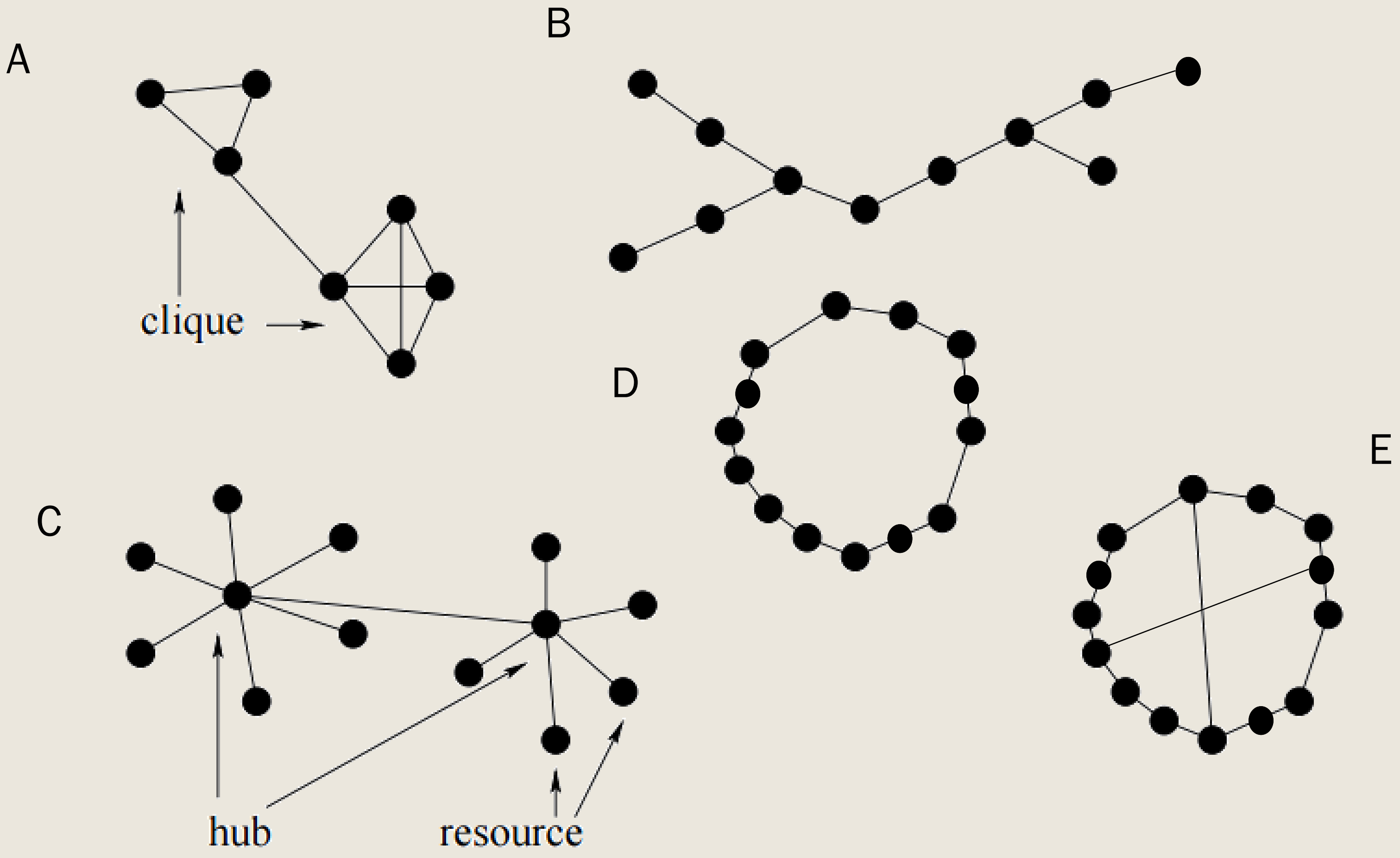
Not every network has a small world effect

- For example: Six degrees only relevant for living (or recent) people.
- Distance between people who lived hundreds of years apart is much larger.

- Similar consideration for "studied with" networks
  - *Example: Carol Kirkwood, BBC weather presenter, graduated from Edinburgh Napier University in 1984.*
  - *How many degrees of separation to you (current students)? Explain your answer.*

Which of these have a small-world effect?

Define small world effect as diameter <= 6



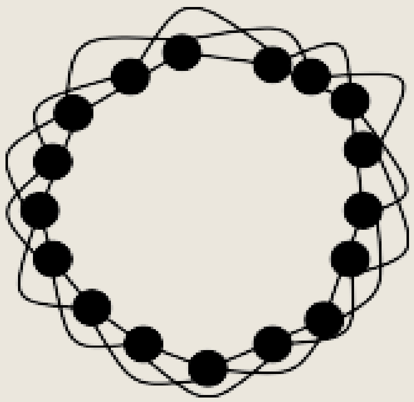
Two types of small world networks

- Watts & Strogatz networks
- Scale-free networks

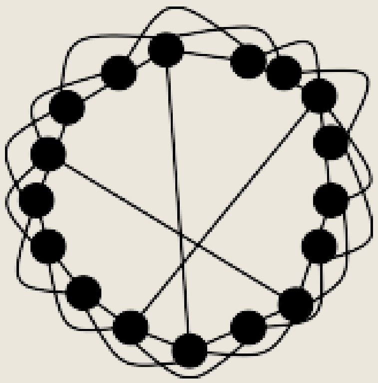
Watts & Strogatz networks

Duncan J. Watts and Steven H. Strogatz, Collective dynamics of small-world networks, Nature, 393, pp. 440–442, 1998

regular structure



Plus some added random links

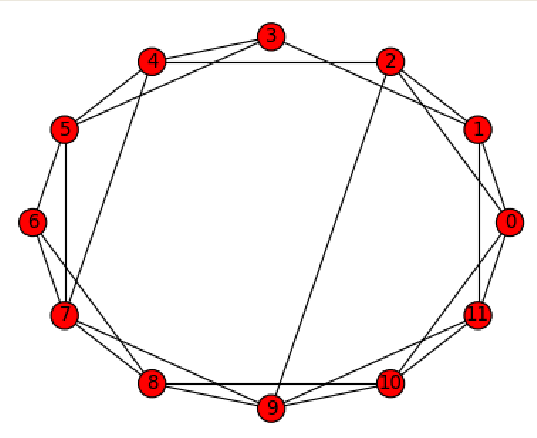


- high local clustering with a few distant links
- egalitarian
- no hubs
- Examples: electric power grid, neuron connections in the brain…

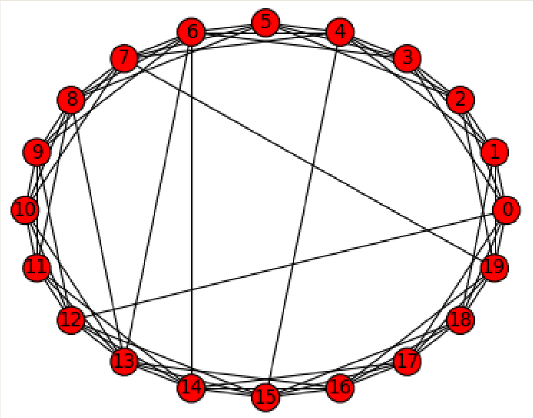
Plus some added random links

Watts & Strogatz networks with networkX

- `watts_strogatz_graph(n,k,p)`
  `newman_watts_strogatz_graph(n,k,p)`
  `connected_watts_strogatz_graph(n,k,p)`
- ring over n nodes;
  each node joined to its k nearest neighbours;
  shortcuts: replacing / adding some edges with probability p.
- See https://networkx.github.io/documentation/latest/reference/generated/networkx.generators.random_graphs.watts_strogatz_graph.html



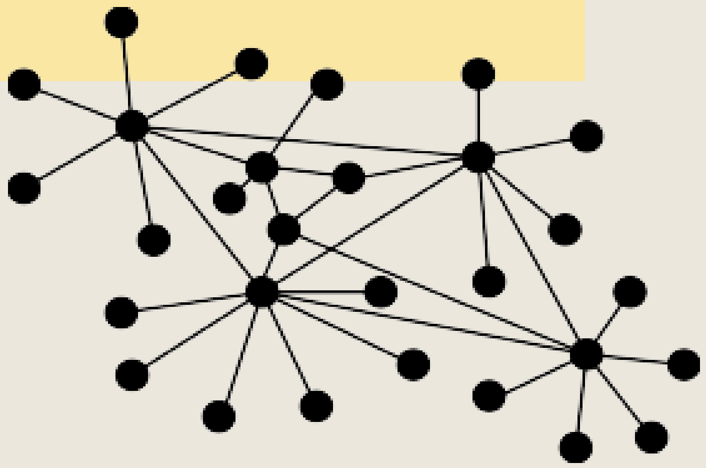
`watts_strogatz_graph(12,4,0.2)`
`[edges replaced - could lead to disconnected graph]`



`newman_watts_strogatz_graph(20,6.0.1)`
`[edges added]`

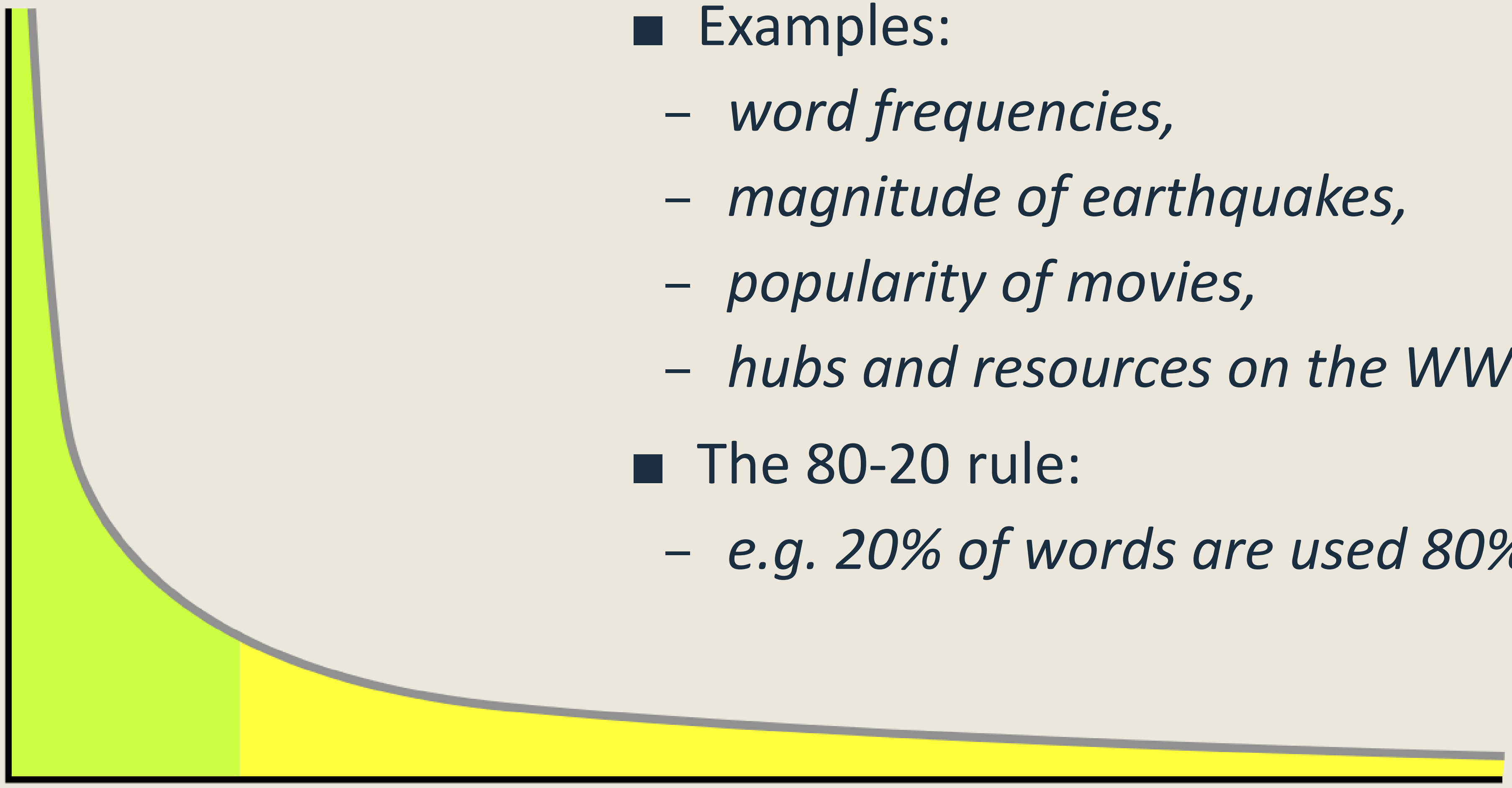
Scripting for Cybersec & Networks

Scale-free networks



- hubs and resources
- power law distribution
- growths: preferential attachment (“the rich get richer”)
- Examples: WWW, citation networks, airline networks, biological networks
- Excellent explanation at https://www.learner.org/courses/mathilluminated/units/11/textbook/05.php
- See also https://en.wikipedia.org/wiki/Scale-free_network#Characteristics

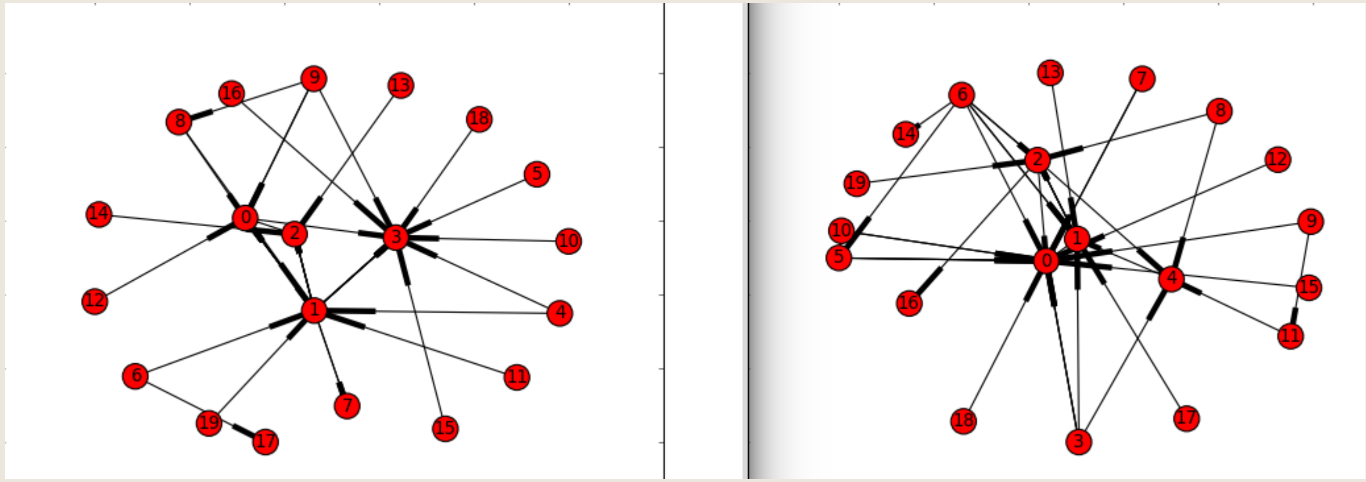
Power law distribution



- Examples:
  - *word frequencies,*
  - *magnitude of earthquakes,*
  - *popularity of movies,*
  - *hubs and resources on the WWW*
- The 80-20 rule:
  - *e.g. 20% of words are used 80% of the time*

Image from https://commons.wikimedia.org/wiki/File:Long_tail.svg

Scale-free graphs in networkX



`scale_free_graph(20)`
[2 different examples, spring layout]

- scale_free_graph(n)
- n is the number of nodes
- Graph is directed
- Lots of options - See https://networkx.github.io/documentation/latest/reference/generated/networkx.generators.directed.scale_free_graph.html

Application: disease control

Diseases are spread by "hubs".
⇒ Outbreaks can be controlled by treating the hubs.

Examples:

- Treatment of rabies in foxes:
  vaccinate the ones that are travelling long distance.
- Influenza: vaccinate children.

Comparison of small world network types

Watts & Strogatz	scale-free
egalitarian no growth local clusters and distant links under attack: deteriorates no particular targets	hubs and resources preferential attachment power law under random attack: stable under targeted attack: weak

Both: small average node-to-node distance

Clustering Coefficient

- The ratio of the number of actual edges there are between neighbours to the number of potential edges there are between neighbours
- In a social network, the probability that two randomly selected friends of A are friends with each other, i.e. the fraction of pairs of A’s friends that are connected to each other.
- The clustering coefficient of a node measures how concentrated the neighbourhood of that node is.
- Network clustering coefficient: the average of all the nodes' coefficients

Clustering coefficient example

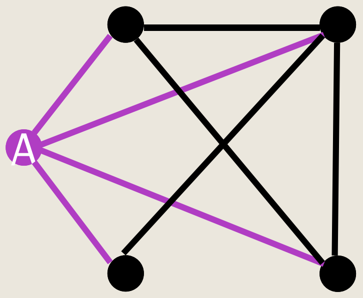
What is the clustering coefficient of node A?

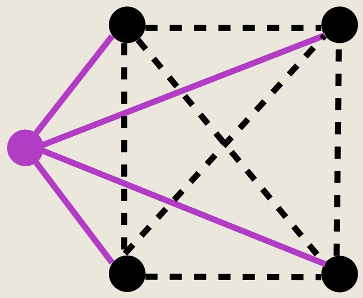
- A has edges to 4 other nodes
- These could have up to $\binom{4}{2} = \frac{4(4-1)}{2} = 6$ edges between them
- They have only 4 of these 6 edges
- Clustering coefficient: 4/6 = 0.67

$\binom{4}{2}$ is "4 choose 2"

4 neighbours

node





4 actual edges among neighbours

$\frac{4 \, x \, 3}{2} = 6$ possible edges among neighbours

Calculating the Clustering Coefficient

Clustering for node n = $\dfrac{|\text{actual edges between neighbours of } n|}{|\text{possible edges between neighbours of } n|}$

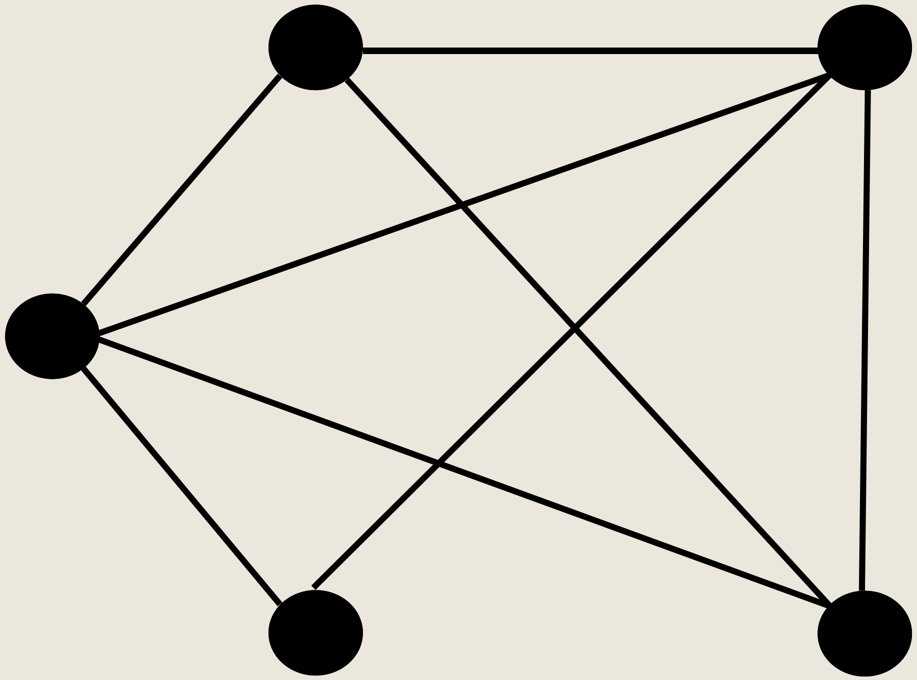
Number of possible edges between k nodes: $\frac{k(k-1)}{2}$
(I.e. number of edges in a complete graph with k nodes)

Clustering coefficient for node n with k neighbours

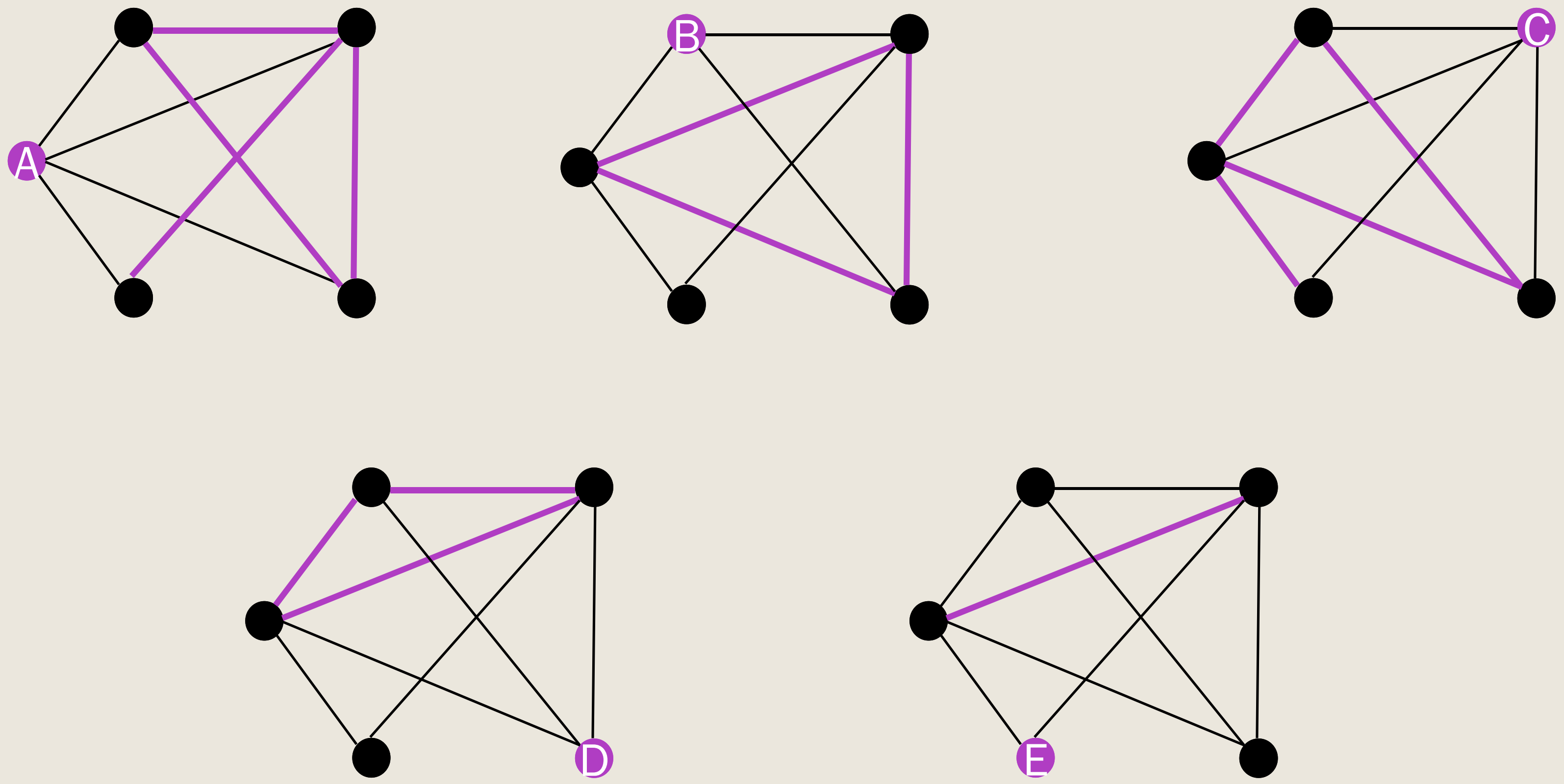
$$C(n) = \frac{|\text{actual edges}|}{\frac{k(k-1)}{2}} = \frac{2 \times |\text{actual edges}|}{k(k-1)}$$

the clustering coefficient for a node with 1 neighbour or with 0 neighbours is 0.

Exercise: Calculate the average clustering coefficient

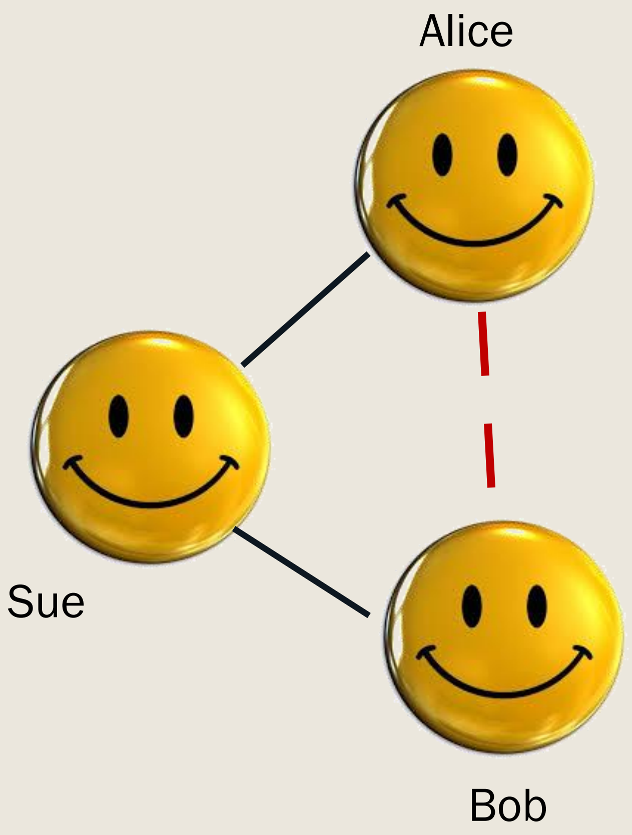


Exercise: Calculate the average clustering coefficient



Triadic closure

- Sue is friends with Alice and with Bob
- According to triadic closure, Alice and Bob are likely to become friends and close the triangle
- if we have 2 snapshots of a social network, the later one will have many new edges that come from this triangle closing operation
- This creates graphs with high clustering coefficients: the small world effect



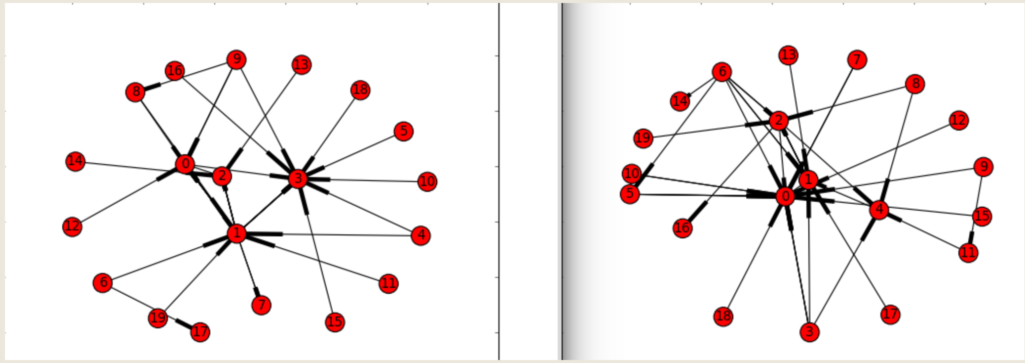
Small-world networks

Properties:

- Exhibit Small-world effect: small average node-to-node distance (shortest path length)
  - *Measured as the **diameter** of a graph*
- Clustering coefficient is larger than the clustering coefficient of a random graph with the same number of nodes and edges.
  - *Having a large clustering coefficient means that "the people you know also know each other".*
  - *This could be the result of triadic closure*

Scale-free graphs in networkX

- scale_free_graph() is not only directed, but also a multi-graph (i.e. a graph with self-loops and multiple edges between the same nodes)
- Cannot calculate clustering coefficient
- A simple graph that looks the same would have a very low clustering coefficient – because there is no triadic closure.
- To convert a scale-free graph SF to an undirected simple graph use SFconv: `SFconv=nx.Graph(SF)`

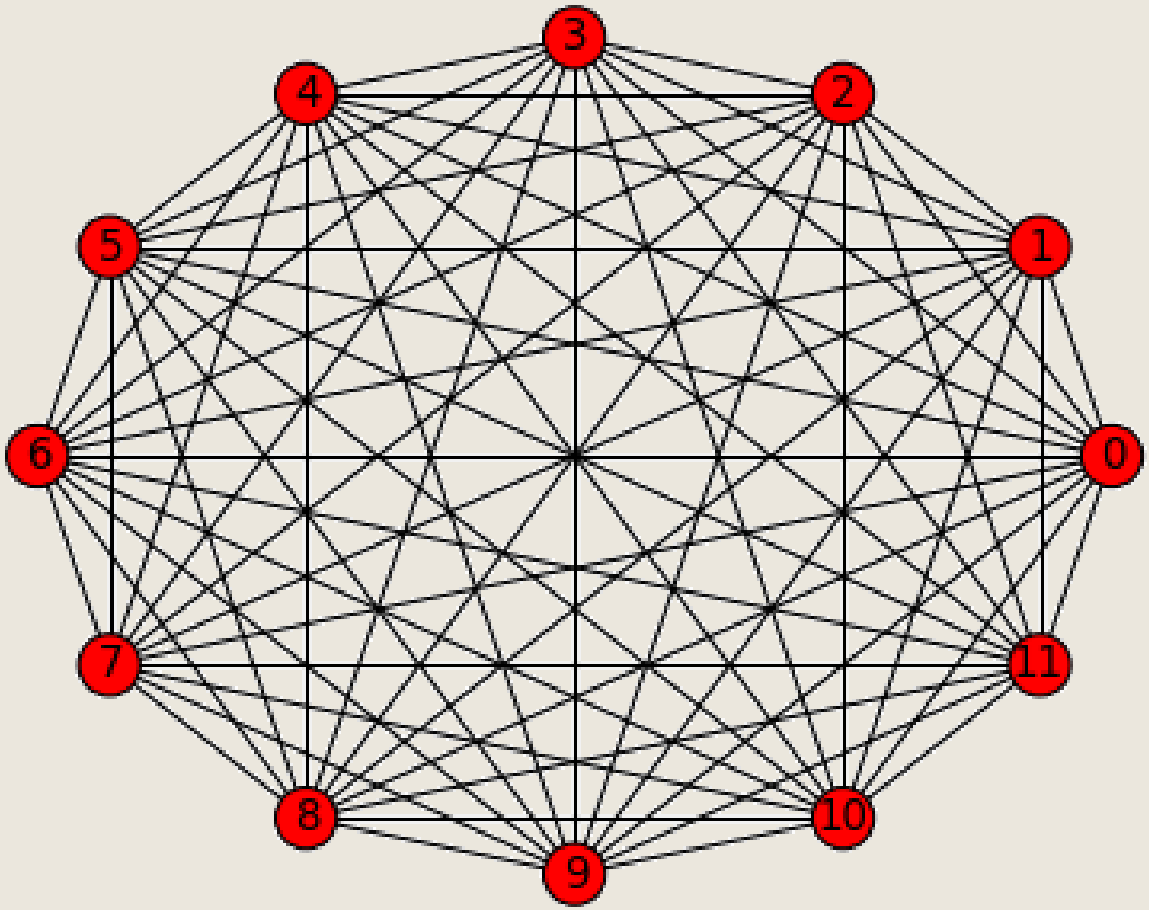


`scale_free_graph(20)`
[2 different examples, spring layout]

Question

For a complete graph

- What is the clustering coefficient?
- What is the diameter?
- Is a complete graph a small world network?



Some Resources

- Interactive network theory tutorial at http://www.learner.org/courses/mathilluminated/interactives/network/. This will take you through various graph and network characteristics and is a great revision of the lecture. (Requires Flash player - works for me in Firefox but not Chrome).
- Highly recommended for labs: NetworkX tutorials (8 page pdf) https://networkx.github.io/documentation/stable/tutorial.html
- Optional: Interactive graph creator http://illuminations.nctm.org/Activity.aspx?id=3550 (the graph explorer tab allows you to test Hamiltonian and Eulerian paths)
- What truth is there in the “6 degrees of separation” theory? Read http://www.theguardian.com/technology/2008/aug/03/internet.email and have a look at http://www.sciencealert.com/are-we-all-really-connected-by-just-six-degrees-of-separation. Additional sources you could have a look at are https://www.psychologytoday.com/articles/200203/six-degrees-urban-myth.
- This lecture is partly based on the book “Small World” by Mark Buchanan.